Programming Task: Information Retrieval System

Akhila Vocakaligara Mani

December 19, 2018

1 Objective

To develop and implement an Information Retrieval System that parses both .txt as well as .html files, indexes the parsed files, performs stemming, ranks the documents using either Vector Space or OkapiBM25 ranking model which can be chosen by the user, and finally searches a given query in the documents.

2 Implementation

Environment: Java - JDK and JRE Version 8
Libraries used

- Apache Lucene: 7.5.0
- Jsoup: 1.11.3

2.1 Code Structure : Functionality of classes

Package used: package com.ext.lucene;

ProcessQueries : This class contains the *main* method, hence execution starts from here. It invokes the method *indexer.createIndex* which is used for indexing.

Indexer : This gives the working of the *indexer.createindex* method. It indexes both .txt and .html files and stores them in the specified location. For indexing, there are two user modes namely:
create : Indexes new files that are not indexed before.
update : Updates the indexed files if any changes are made to the original document.

MyStemmingAnalyzer : This class extends the *Analyzer* class for implementation of Porter Stemmer. The document is first tokenized using *Analyzer.TokenStreamComponents*.

Then the filters namely PorterStemmer, LowerCase filter and Synonym filter are implemented.

SearchFields : The class is used to search multiple fields concurrently. This is done using a properties.txt file which avoids the recompilation of classes. The file contains fields separated by comma with the last field ending with a comma.

MyQueryParser : Searches the query from the indexed files by using the method *searchIndexWithQueryParser* and also ranks the documents based on the ranking model[VS/OK]. In addition it validates wildcard queries.

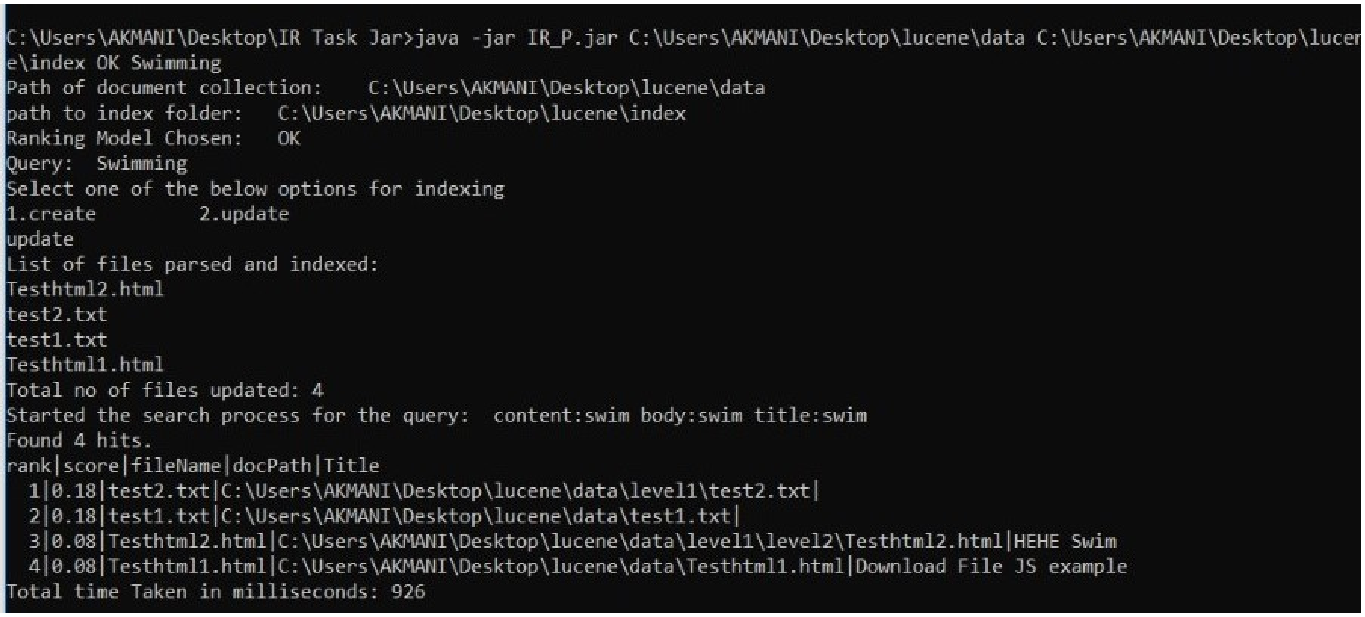
LuceneConstants : Contains all the constants used for the overall implementation.

Sample command for executing jar file(file name containing spaces)

```
java -jar IR_P.jar C:\Users\AKMANI\Desktop\IR" Task "Jar\data C:\Users\AKMANI\Desktop\
IR" Task "Jar\index OK Swimming in water
```

3 Results

The Information Retrieval System successfully ranks the 10 most relevant documents by displaying the rank, score, file name, file path and title(in case of .html file) of each of the documents.



```
C:\Users\AKMANI\Desktop\IR Task Jar>java -jar IR_P.jar C:\Users\AKMANI\Desktop\lucene\data C:\Users\AKMANI\Desktop\lucen
e\index OK Swimming
Path of document collection:    C:\Users\AKMANI\Desktop\lucene\data
path to index folder:   C:\Users\AKMANI\Desktop\lucene\index
Ranking Model Chosen:   OK
Query:  Swimming
Select one of the below options for indexing
1.create         2.update
update
List of files parsed and indexed:
Testhtml2.html
test2.txt
test1.txt
Testhtml1.html
Total no of files updated: 4
Started the search process for the query:  content:swim body:swim title:swim
Found 4 hits.
rank|score|fileName|docPath|Title
  1|0.18|test2.txt|C:\Users\AKMANI\Desktop\lucene\data\level1\test2.txt|
  2|0.18|test1.txt|C:\Users\AKMANI\Desktop\lucene\data\test1.txt|
  3|0.08|Testhtml2.html|C:\Users\AKMANI\Desktop\lucene\data\level1\level2\Testhtml2.html|HEHE Swim
  4|0.08|Testhtml1.html|C:\Users\AKMANI\Desktop\lucene\data\Testhtml1.html|Download File JS example
Total time Taken in milliseconds: 926
```

Figure 1: Output